

# VIDEO USER INTERFACE SYSTEM AND METHOD

## CROSS REFERENCE TO RELATED APPLICATIONS

5 [0001] The present application claims the benefit of the filing date of  
U.S. Provisional Application Serial No. 60/502,728, filed September 12,  
2003, entitled "Video User Interface System and Method," the entire  
teachings of which are incorporated herein by reference. This application  
is related to co-pending U.S. Application Serial No. 10/\_\_\_\_\_, filed  
March 17, 2004, entitled "Imaging System and Method for Displaying  
10 and/or Recording Undistorted Wide-Angle Image Data."

## BACKGROUND OF THE INVENTION

### *Field of the Invention*

15 [0002] This invention generally relates to imaging systems, and more  
particularly, to a system and method for generating a graphical user  
interface from image data, such as video images.

### *Background Description*

20 [0003] Conventional video surveillance systems allow users to view  
portions of monitored areas by either physically aiming a video camera at  
a desired portion or by generating a perspective corrected view of the  
desired portion from a wide angle (e.g., fisheye) image of the monitored  
area. One shortcoming of such conventional systems is that they do not  
conveniently enable a user to discern the part of the monitored area  
represented by the displayed portion. For example, a video surveillance  
25 system may display a close-up image of a person or an object (the  
displayed portion) in a room or a parking lot (the monitored area).  
Without an image of the entire room or lot and an indication of the area

represented by the close-up image, it may be difficult or impossible to quickly determine the position of the person or object in the monitored room or lot. This is especially true for a crowded room or lot, where it may be difficult to distinguish one isolated portion from another.

5     **[0004]** Another shortcoming is that video surveillance systems typically do not allow an operator to see anything but the portions of monitored areas being displayed. The undisplayed portions may comprise a substantial portion of a monitored area, in which significant activity of interest may occur. For example, if a displayed image represents only  
10     10% of a monitored parking lot, the other 90% may be unmonitored at any given time.

**[0005]** Yet another shortcoming is that high zoom factor images mathematically derived from wide-angle lens image data tend to exhibit low resolution. The greater the zoom factor, the lower the resolution.  
15     This impairs an operator's ability to distinguish fine details (e.g., facial features or automobile license plate numbers) in a monitored area.

**[0006]** While the invention overcomes one or more of the problems in conventional video surveillance systems as set forth above, it should be apparent that the invention may have applicability to a wide array of  
20     imaging devices and methods, particularly those employing images of moving objects, regardless of the particular format of the image data.

## **SUMMARY OF THE INVENTION**

**[0007]** The invention avoids the drawbacks of the prior art by providing  
25     a graphical user interface that may, according to one aspect of the invention, include one or more panoramic views of a monitored area, one or more virtual views preferably corresponding to portions of the monitored area, and/or reference data such as a reference window overlaid

in a panoramic view to identify a portion of the panoramic view corresponding to a virtual view. Panoramic views of the invention, which may be derived from wide-angle lens image data, exhibit a determined (e.g., upright) orientation and substantially reduced (or no) wide-angle image distortion, thus providing a readily discernible view. Virtual views of the invention, which may be based upon either transformed wide-angle lens image data or image data from a camera system without a wide-angle lens, also exhibit substantially reduced (or no) wide-angle image distortion, thus providing a readily discernible view. The virtual views and reference windows may respond to operator commands.

**[0008]** In particular, according to a first aspect of the invention, a system is provided for creating signals indicative of a graphical user interface from wide-angle image data, such as video images, corresponding to a monitored area. The system includes a buffer configured to receive wide-angle image data such as video images, corresponding to the monitored area. A processor operably coupled to the buffer is configured to transform wide angle image data received by the buffer into panoramic view data corresponding to at least one panoramic view of the monitored area, and into virtual view data corresponding to at least one virtual view of a portion of the at least one panoramic view.

**[0009]** In another aspect of the invention, a method of creating a graphical user interface is provided. The method includes the steps of buffering wide-angle data corresponding to a wide-angle image, such as a video image, of a monitored area; transforming the buffered data into panoramic data for at least one panoramic view using a panoramic transformation; and transforming portions of the buffered data into virtual view data for at least one virtual view using a virtual view transformation.

[0010] Additionally, the graphical user interface may include a reference window overlaid in a panoramic view to identify a portion of the panoramic view represented by a virtual view. As another option, the system may include a dome camera system configured to provide image data for a virtual view according to commands. The dome camera image data may exhibit higher resolution than the wide-angle data.

[0011] In yet another aspect of the invention, a system for creating signals indicative of a graphical user interface from wide-angle image data corresponding to a monitored area is provided. The system includes a means for storing wide-angle image data corresponding to a monitored area. The system also includes a means for processing and transforming data received from said storing means into panoramic view data corresponding to at least one panoramic view of the monitored area and into virtual view data corresponding to at least one virtual view of a portion of the at least one panoramic view.

[0012] Additional features, advantages, and embodiments of the invention may be set forth or apparent from consideration of the following detailed description, drawings, and claims. Moreover, it is to be understood that both the foregoing summary of the invention and the following detailed description are exemplary and intended to provide further explanation without limiting the scope of the invention as claimed.

#### **BRIEF DESCRIPTION OF THE DRAWINGS**

[0013] The accompanying drawings, which are included to provide a further understanding of the invention, are incorporated in and constitute a part of this specification, illustrate preferred embodiments of the invention and together with the detail description serve to explain the principles of the invention. In the drawings:

[0014] Figure 1 is a block diagram that conceptually illustrates a graphical user interface constructed in accordance with the principles of the invention;

[0015] Figure 2 is a block diagram that conceptually shows an imaging system for producing a graphical user interface in accordance with an exemplary embodiment of the invention;

[0016] Figure 3 conceptually shows  $\theta$  (i.e., the pan rotation angle around the Z-axis) and  $\phi$  (the tilt rotation angle around the X-axis) for a transformation in accordance with an exemplary implementation of the invention;

[0017] Figure 4 conceptually shows image center IC<sub>xy</sub> based on image radii IR<sub>x</sub> and IR<sub>y</sub> in accordance with an exemplary implementation of the invention;

[0018] Figure 5 is a high level flowchart conceptually illustrating steps of a methodology for creating a graphical user interface in accordance with an exemplary implementation of the invention; and

[0019] Figure 6 is a high level flowchart conceptually illustrating steps of a methodology for creating a graphical user interface including using dome camera system positioning in accordance with an exemplary implementation of the invention.

## DETAILED DESCRIPTION

### *User Interface*

[0020] An exemplary system in accordance with the invention includes a processor configured to transform digital image data corresponding to the wide-angle images, such as video images, of a monitored area into data corresponding to one or more panoramic views of the monitored area. Referring to **Figure 1**, a user interface in accordance with an exemplary

implementation of the invention is conceptually shown. Strips **110** and **160**, which may be rectangular in shape as illustrated, correspond to transformed panoramic views. The transformed panoramic views **110** and **160** exhibit upright orientation and substantially reduced (or no) wide-angle image distortion, thus providing a readily discernible view of the monitored area. Examples of a suitable transformation carried out by the processor are discussed subsequently.

**[0021]** The processor is also configured to provide data corresponding to one or more virtual views **130** and **140**, each of which is an undistorted (or substantially undistorted) view of a user-determined portion of the panoramic views **110** and **160**. Virtual view data may be obtained either by transformation of portions of the wide-angle video image data, or by positioning and focusing a camera having a normal lens (i.e., not a wide angle lens) or a telephoto lens to produce image data for the determined portion of the monitored area. Again, a suitable transformation to obtain the virtual view data is discussed subsequently.

**[0022]** The processor may be further configured to provide reference data (e.g., windows - i.e., graphically generated boxes or other shapes overlaid in the panoramic views - boxes **120** and **150** as shown in **Figure 1**) corresponding to the undistorted view of the determined portions of the panoramic views represented by the virtual views **130** and **140**. Thus, the reference data **120** and **150** reveals what portions of the panoramic views are represented by the virtual views **130** and **140**.

**[0023]** As an operator pans, tilts or zooms either of the virtual views **130** and **140** to correspond to any part of either of panoramic views **110** and **160**, the corresponding reference window **120** or **150** overlaid on top of a panoramic view **110** and **160** moves in the panoramic view **110** or **160**,

thus indicating the part of the panoramic view 110 and 160 represented by the virtual view 130 or 140.

5 [0024] The system is preferably configured to allow a user to select which virtual view 130 or 140 to currently control. The active (i.e., currently controlled) virtual view 130 or 140 may be distinguished from the inactive virtual view by an indicia. For example, a white frame may surround the active virtual window and a black frame may surround the inactive virtual view. The corresponding reference window 120 or 150 may also be highlighted in either black or white based on its associated  
10 virtual view's active/inactive status.

[0025] The exemplary display screen may have a resolution of 486 by 720 pixels. The exemplary panoramic views, as shown in **Figure 1**, are each 720 pixels wide and 120 pixels high. The exemplary virtual views, as shown in **Figure 1**, may each be 360 pixels wide and 246 pixels high.  
15 Those skilled in the art will appreciate that the invention is not limited to the exemplary resolution or view sizes. Many other resolutions and sizes are feasible and come within the scope of the invention.

[0026] The reference data (e.g., reference windows 120 and 150) may be controllable via a user input device, such as a joystick, cursors or other  
20 pointing device. Thus, for example, a user may move and resize reference windows 120 and 150 to cover any portions of the panoramic views 110 and 160. The portions of the panoramic views covered by the reference windows 120 and 150 correspond to or define the virtual views 130 and 140. The system may also automatically determine the portions of the  
25 panoramic views 110 and 160 to cover, such as by one or more determined algorithms (e.g., move reference windows as a function of time) or in response to one or more signals (e.g., a motion detector, gate sensor, or alarm signal). Other manifestations of reference data, such as numerical

coordinates or cross hairs, may be used in lieu of or in addition to reference windows **120** and **150**.

**[0027]** Panoramic views **110** and **160** and virtual views **130** and **140** may be displayed concurrently on a display monitor, thus providing a readily discernible view of the entire monitored area and an undistorted (or substantially undistorted) view of a portion of the monitored area. The reference data **120** and **150** may also be displayed with the panoramic views **110** and **160**, thus enabling a user to readily identify the portions of the panoramic views **110** and **160** that are represented by the virtual views **130** and **140**.

**[0028]** Those skilled in the art will appreciate that the invention is not limited to the exemplary number, size, shape or arrangement of panoramic views and/or reference windows shown in **Figure 1** and discussed above. For example, one or more panoramic views **110** and **160**, virtual views **130** and **140** and or reference windows **120** and **150** may be provided without departing from the scope of the invention. Additionally, the user interface components may be arranged as shown in **Figure 1** and discussed above, or differently, on one or more display monitors, or in one or more windowed portions of one or more display monitors, without departing from the scope of the invention. Furthermore, the interface may be configured to allow a user to control the display arrangement. For example, a user may hide or move a panoramic view **110** or **160**, and/or a virtual view **130** or **140** and/or a reference window **120** or **150**; or hide all virtual views **130** and **140** reference windows **120** and **150**; or opt to display only a selected panoramic view **110** or **160** or virtual view **130** or **140**.



### *Hardware*

[0029] Various image sources may supply image data for use by a system in accordance with the invention. By way of example and not limitation, one or more imagers such as digital or analog video cameras may be used to supply video data, which may be in any known format. Thus, unless otherwise apparent from the context, as used herein “video” broadly refers to images, such as images of stationary and/or moving objects, which may or may not be produced using signals in standard formats, such as NTSC signals. The system of the invention may operate in real time, processing video data as it is produced using the video imagers and without appreciably delaying output to a display monitor or recorder. Alternatively, saved analog or digital video signals or data may serve as the source of images.

[0030] Referring now to **Figure 2**, a high-level block diagram of a video imaging system **200** for use with a system in accordance with an exemplary implementation of the invention is shown. The video imaging system **200** includes a video camera **210** equipped with a wide-angle lens **205**, such as a fisheye lens. Various wide-angle lenses are available, many of which, for example, have an angular field of view of approximately 50 degrees or more. The wide-angle lens may preferably be suitable for producing an image of an entire area to be monitored (e.g., portions of a room excluding the ceiling) when the wide-angle lens is positioned at a determined point relative to the monitored area (e.g., on a ceiling of the room, at or near the center of the ceiling, pointing downwardly). By way of example and not limitation, a fisheye lens, which is a type of wide-angle lens, and which may have a field of view of 180 degrees, may be utilized. The camera **210** may be a digital camera such as a camera that employs a charge coupled device (CCD) array, a Time Delay Integration

(TDI) CCD array, complementary metal oxide semiconductor (CMOS) image sensors or other sensors or devices for electronic image production. Alternatively, the camera 210 may be an analog camera that produces analog (e.g. NTSC or PAL) output signals.

5     **[0031]**     The wide-angle lens 205 is aimed at the area to be monitored. The lens 205 may be a fisheye lens or another type of wide-angle lens that provides a 180° field of view, or a wider or narrower field of view. The position and orientation may generate a fisheye view of the entire monitored area.

10    **[0032]**     A video decoder 215 operably coupled to the camera translates analog signals from the camera into digital video data. The decoder 215 may include a clock circuit (e.g., a 27 MHz video clock circuit) suitable for synchronization and processing of video image data streams. Clock signals 217 from the clock circuit may be transmitted to processor 225.

15    **[0033]**     By way of example and not limitation, a suitable decoder 215 may include commercially available decoders, such as an SAA7113H 9-bit video input processor available from Phillips Electronics N.V. The SAA7113H 9-bit video input processor includes an anti-aliasing filter, an automatic clamp and gain control, a clock generation circuit, a digital  
20    multi-standard decoder, and a brightness, contrast and saturation control circuit. It is able to decode Phase Alternation Line (PAL), French Sequential Couleur avec Memoire (SECAM) and American National Television Systems Committee (NTSC) standard signals into CCIR-601 (International Radio Consultative Committee, now the International  
25    Telecommunication Union Radiocommunication Sector, standard for encoding analog video signals in digital form) compatible color component data values. The SAA7113H accepts as analog inputs CVBS (i.e., composite Chroma Video Blanking & Sync inputs) or S-video (Y/C).

[0034] Digital video data from decoder **215** may be stored in an image buffer **220**. The image buffer **220** provides a means for storing wide-angle image data corresponding to a monitored area. The image buffer **220** may be comprised of volatile or non-volatile memory or other devices  
5 configured to store video image data. The data may be stored temporarily, for example, until no longer needed for transformation by the image data processor **225**, as described below, until new video image data is required to be stored in the occupied portion of the buffer, or until some other event occurs or a period of time elapses. The buffer **220** may be sized to store  
10 image data corresponding to an entire field or frame of video, or more or less data.

[0035] One or more user input modules **290** may also be provided to generate user commands. By way of example, the module may include an input device **275** such as a keypad, digitizer, joystick, microphone and  
15 voice recognition module, or some other device configured to enable a user to enter commands. The input device **275** may be operably coupled to an input processor **280** and input memory **285** configured to produce user command data **292** corresponding to user input commands. The input processor **280** may be implemented as a special purpose computer; a  
20 programmed general purpose computer; a programmed microprocessor, microcontroller or programmable read-only memory and peripheral integrated circuit elements; an application specific integrated circuit (ASIC) or other integrated circuits; a digital signal processor; a hardwired electronic or logic circuit such as a discrete element circuit; a  
25 programmable logic device such as a field programmable gate array (FPGA), or the like. The input memory **285** may include volatile or non-volatile memory such as, for example, static random access memory (SRAM), dynamic random access memory (DRAM), fast page mode

dynamic random access memory (FPM DRAM), extended data-out  
dynamic random access memory (EDO DRAM), synchronous dynamic  
random access memory (SDRAM), double data rate synchronous dynamic  
RAM (DDR SDRAM), electronically erasable programmable read only  
5 memory (EEPROM) such as flash memory with or without a controller,  
hard disk enabled virtual memory, and/or other data storage devices that  
may be operably coupled to the input processor **280**. Using a user input  
device **275**, a user may identify a portion of a panoramic view to be  
displayed as a virtual view. A user may also control zoom factors for  
10 virtual views, and other display parameters via the user input device **275**.  
[0036] An image data processor **225** may receive buffered image data  
**222**, clock signals **217** and user command data **292** to produce panoramic  
view data, virtual view data and reference data. The image data processor  
**225** thus provides a means for processing and transforming data received  
15 from the storing means (i.e., the image buffer **220**) into panoramic view  
data corresponding to at least one panoramic view of the monitored area  
and into virtual view data corresponding to at least one virtual view of a  
portion of the at least one panoramic view. The processor **225** may be  
implemented as a special purpose computer; a programmed general  
20 purpose computer; a programmed microprocessor, microcontroller or  
programmable read-only memory and peripheral integrated circuit  
elements; an application specific integrated circuit (ASIC) or other  
integrated circuits; a digital signal processor; a hardwired electronic or  
logic circuit such as a discrete element circuit; a programmable logic  
25 device such as a field programmable gate array (FPGA), or the like. A  
preferred implementation uses an FPGA or an ASIC as the processor **225**.  
When a system design is considered stable, an FPGA could be replaced  
with an ASIC to reduce system cost.

[0037] The image data processor 225 may also be operably coupled to memory 230, which may, for example, be non-volatile memory in the form of electronically erasable programmable read only memory (EEPROM), such as flash memory, with or without a controller; a solid-state floppy-disk card (SSFDC) such as SmartMedia<sup>®</sup> memory commercially available from Kabushiki Kaisha Toshiba, Toshiba Corporation, and Toshiba America Electronic Components, Inc.; CompactFlash<sup>®</sup> memory by SanDisk Corporation; hard disk enabled virtual memory; and/or other non-volatile storage devices. Illustratively, memory 230 may be used to facilitate processor computations by storing one or more tables of values for frequently calculated functions, especially functions that might otherwise consume significant processor 225 resources. By way of example and not limitation, memory 230 may store a table of values, such as cosine, sine, 1/cosine, 1/sine, or other values based on trigonometric or other types of functions used in an applicable transformation algorithm, as discussed more fully below. The indices for the table may be angular values in degrees, radians, other types of measurements, or some other possible value for a variable. The table entries may be pre-calculated values based on the corresponding indices. Trigonometric functions and complex floating point calculations can otherwise be a time and resource consuming part of processor 225 calculations. The use of pre-determined stored values conserves processor resources and may lead to an appreciable increase in overall computation speed.

[0038] As discussed more fully below, the image data processor 225 mathematically transforms wide angle video image data in the image buffer 220 into panoramic views using a panoramic view transformation algorithm. Depending upon the implementation, the image data processor

**225** may also transform wide angle video image data in the image buffer **220** into virtual views using a virtual view transformation algorithm, as discussed more fully below.

5     **[0039]** Upon transformation, the processor **225** sends transformed video data to an encoder **235**. The encoder converts the digital video data into output signals compatible with an output device such as a display monitor **240** and/or video recorder **245**.

10     **[0040]** By way of example and not limitation, the encoder may be any suitable commercially available decoder, such as an SAA7121H digital video encoder by Phillips Electronics, N.V. The SAA7121H digital video encoder circuit accepts CCIR compatible YUV data with 720 active pixels per line and encodes the digital YUV video data to NTSC, PAL, CVBS or S-video signals.

15     **[0041]** Output from the encoder **235** is sent to an output device, such as a standard display monitor **240** and/or video recorder **245**. The display monitor may be any device configured to visually display images based on electronic video signals output from the encoder **235**. The recorder may be any device configured to record video signals from the encoder **235** on removable or non-removable storage media such as, for example, a  
20     magnetic tape, diskette, CD-ROM, DVD, hard disk, memory (e.g., nonvolatile EEPROM) or the like. If both a display monitor **240** and recorder **245** are used, the devices may be configured in parallel or in series (e.g., without output from the video recorder being sent to a display monitor).

25     **[0042]** In an alternative embodiment, instead of transforming wide angle video image data in the image buffer **220** into a virtual view using a virtual view transformation algorithm, the image data processor **225** may generate

virtual view data from video image data produced by a one or more additional camera systems **250**. An additional camera system may include an analog or digital camera **260** with a normal lens **255** (i.e., not a wide angle lens) aimed at a determined portion of the monitored area. The lens  
5 **255** may have a narrower Field of View ( FOV), e.g., 45°, which would provide higher resolution than wide-angle lens **205**, thus providing a more detailed video image for a virtual view. Thus, the additional camera enables switching from a low resolution virtual view derived from wide-angle lens image data to a high resolution virtual view, with high zoom  
10 factor capability, produced using the additional camera system. This gives the user the ability to isolate small details of a monitored area.

[0043] The additional camera system **250** may also include a video decoder **265** that is operably coupled to the camera and configured to translate analog signals from the camera into digital video data. The  
15 decoder **265** may also include a clock circuit (e.g., a 27 MHz video clock circuit) suitable for synchronization and processing of video image data streams. Clock signals **267** may be transmitted to processor **225**.

[0044] The additional camera system **250** may further include an image buffer **270**. The image buffer **270** may be comprised of volatile or non-  
20 volatile memory or other devices configured to temporarily store video image data, such as, for example, static random access memory (SRAM), dynamic random access memory (DRAM), fast page mode dynamic random access memory (FPM DRAM), extended data-out dynamic random access memory (EDO DRAM), synchronous dynamic random  
25 access memory (SDRAM), double data rate synchronous dynamic RAM (DDR SDRAM), Rambus dynamic random access memory (RDRAM), multiport dynamic random access memory (MPDRAM), synchronous graphics RAM (SGRAM), electronically erasable programmable read only

memory (EEPROM), hard disk enabled virtual memory and/or other data storage device. The buffer 270 may be sized to store image data corresponding to an entire field or frame of video, or more or less data. Buffered video data 272 may be transmitted to processor 225. That data  
5 may then be processed and encoded 235 for display in either virtual view 130 or 140.

[0045] Moreover, the additional camera system 250 may be configured as a programmable dome camera system. Dome cameras can typically rotate (i.e., pan) 360 degrees, and tilt and zoom according to control  
10 signals. The dome camera 260 may be coupled to a processor programmed to produce control signals causing the camera 260 to aim at particular portions of a monitored area in particular patterns. Alternatively, processor 225 may produce control data 294, based upon user command data 292 from user input module 290, to control  
15 positioning and movement of the dome camera. A control interface (not shown) may also be operably coupled to dome camera 260 and processor 225, as necessary to convert control data output from the processor 225 into signals for controlling dome camera 260 position, movement and zooming.

#### 20 *Transformation Methodologies*

[0046] Referring again to Figure 1, panoramic views 110 and 160 and virtual views 130 and 140 may be produced by transforming wide-angle image data (including fisheye image data) into 2-dimensional space, without distortion (or with substantially reduced distortion). Many such  
25 transformation processes (also referred to as dewarping, mapping, remapping, planar projection, conversion and perspective correction) are known in the art and can be adapted, in accordance with the teachings herein, to produce panoramic views 110 and 160 and virtual views 130



and 140 with linear perspective geometry from wide-angle image data (including fisheye image data). While the invention is not limited to any specific transformation processes, an exemplary panoramic view transformation and an exemplary virtual view transformation are described more fully below for use with a system of the invention employing fisheye lens image data.

[0047] The panoramic view transformation process involves mathematically transforming wide angle video image data into 2-dimensional space for producing panoramic views. The exemplary transformation entails determining, for each output window pixel coordinate, an address within the image buffer 220 for pixel data to be encoded and output to a display 240 and/or recorder 245. In a preferred implementation, this address calculation is timed such that the calculated address for each output pixel is ready just in time to feed the output encoder. This implementation advantageously eliminates the need for an output image buffer.

[0048] In an exemplary implementation, two panoramic views 110 and 160 are produced. One panoramic view 110 corresponds to half of the monitored area (e.g., a 0-180 degree panoramic view strip). The other panoramic view corresponds to the other half of the monitored area (e.g., a 181-360° panoramic view strip). Thus, in this implementation, the second panoramic view is offset from the first panoramic view by half of the field of view. The points dividing one half from the other half may be pre-programmed, user configurable or user controllable. The two panoramic views combined provide a full (e.g., a 360°) view of the monitored area.

[0049] The exemplary panoramic views are 720 pixels wide and 120 pixels high. Output window pixel coordinates range from -360 to +360 for **W<sub>x</sub>** and from -60 to +60 for **W<sub>y</sub>**. The **W<sub>x</sub>** and **W<sub>y</sub>** coordinates are

derived from the output view column and line pixel counters, and together with the desired Zoom factor, are transformed into  $\theta$  (i.e., the pan rotation angle around the Z-axis) and  $\phi$  (the tilt rotation angle around the X-axis), as conceptually illustrated in **Figure 3**. This information is then converted to an address in the input capture buffer. The data from this address is output to the encoder for display. The variables Pan, Tilt and Zoom may be programmed or commanded via user input module **290**.

Given:

$$\text{Zoom\_1X\_Tilt\_FOV} = 45^\circ$$

$$\text{ScaleFactorX(SFx)} = 180^\circ/720 \text{ pixels} = 0.25^\circ/\text{pixel}$$

$$\text{Zoom Scale(ZS)} = 1 \text{ (Requested zoom factor ie: 1, 2, 3 etc.)}$$

$$\text{ScaleFactorY(SFy)} = (\text{Zoom\_1X\_FOV}) \times (1/\text{ZS}/120\text{pixels})$$

$$\text{ScaleFactorY(SFy)} = 45^\circ/120 \text{ pixels} = 0.375^\circ/\text{pixel}$$

$$\text{Pan} = \text{Requested Pan angle in degrees}$$

$$\text{Tilt} = \text{Requested Tilt angle in degrees}$$

$$\text{Wx} = \text{Output Window pixel x coordinate (-360 to +360)}$$

$$\text{Wy} = \text{Output Window pixel y coordinate ( +60 down to -60)}$$

$$\text{IRx} = \text{Image buffer fisheye image X axis radius in pixels}$$

$$\text{IRy} = \text{Image buffer fisheye image Y axis radius in pixels}$$

$$\text{ICx} = \text{Image buffer fisheye image center X coordinate in pixels, as conceptually shown in Figure 4}$$

$$\text{ICy} = \text{Image buffer fisheye image center Y coordinate in pixels, as conceptually shown in Figure 4.}$$

**[0050]** The horizontal component of the panoramic view (starting with output window upper left at **-360,+120**) is converted from pixel units to degrees(**SFx**). Each output pixel equates to .25 degrees (**180°/720 pixels**). The programmable **Pan** value is added to each pixel value. The second panoramic view **160** is offset from the first **110** by **+180** degrees.

Therefore:

**Equation 1**  $\theta = [(Wx + 360) \times SFx] + Pan = [(Wx + 360) \times 0.25] + Pan$

**Equation 2**  $\theta = 0.25Wx + 90 + Pan$

5 **[0051]** The vertical component of the panoramic view (starting with output window upper left at origin -360,+120) is converted from pixel units to degrees (SFy). Each vertical pixel equates to 0.375 degrees (45°/120lines). The programmable Tilt value is added to each pixel value. The programmable zoom performs a scale (45° x (1/zoom)) / 120. Therefore, Zoom Scale equals 1.

10 **Equation 3**  $\phi = (\pi/2 - Tilt) - [(Wy + 60) \times (SFy)]$

**Equation 4**  $\phi = 90 - Tilt - 0.375Wy - 22.5$

**Equation 5**  $\phi = 67.5 - (Tilt + 0.375Wy)$

**Equation 6**  $Radius = -\phi / \pi/2$

**Equation 7**  $Radius = -\phi / 90$

15 **[0052]** Next, the input image X,Y (Ixy) pixel is calculated, as follows:

**Equation 8**  $Ix = (Radius \times \cos(\theta) \times IRx) + ICx$

**Equation 9**  $Iy = (Radius \times \sin(\theta) \times IRy) + ICy$

20 **[0053]** The image buffer 220 is preferably addressed as a 1024 byte by 1024 byte array. The memory location, in the image buffer for the pixel data is calculated by:

**Equation 10**  $PixelData = (Iy \times 1024) + Ix$

**[0054]** PixelData may then be sent to the encoder 235 for output to the output device such as a display monitor 240 and or a recorder 245. Pixel Data for each pixel comprising the panoramic views 110 and 160 may be  
25 determined in a similar manner.

[0055] Each virtual view window 130 and 140 provides a perspective corrected view (e.g., a view with no, or substantially reduced, fisheye distortion) of any portion of a panoramic view 110 or 160. The virtual view windows are preferably 360 pixels wide and 240 pixels high. The center of the window is defined as 0,0. **Wx** values range from -180 to +180 and **Wy** values range from +120 down to -120. The **Wx** and **Wy** coordinates are taken from the output window line and pixel counters and, together with the desired Pan, Tilt and Zoom Scale, are transformed into  $\theta$ ,  $\phi$ , and Radius using the virtual view transformation algorithm. The processor determines which algorithm (i.e., the virtual view transformation algorithm or the panoramic view transformation algorithm) to use based on where the counters indicate it is in the output video frame. As with the panoramic views,  $\theta$  is defined as the Pan angle around the Z-Axis, where the positive Y direction, as shown in Figure 3, is 0 degrees. Additionally,  $\phi$  is the Tilt angle around the X-Axis where the positive Z direction, pointing straight down as shown in Figure 3, is 0 degrees. Z is the perpendicular distance from the X-Y plane down to the surface of a unity radius hemispheric fisheye model.

[0056] The virtual view transformation algorithm entails converting the X and Y output window coordinates to an equivalent proportion of a unity radius hemisphere model using a current Zoom factor and a measured radius-X and radius-Y from an input fisheye circular image. These adjusted values are used to calculate a corresponding Z value. The X, Y and Z coordinates are rotated around the unity radius hemisphere model using the commanded Pan and Tilt from the user input module 290. The rotated X and Y values are then rescaled back to pixels and lines, based on measured radius-X and radius-Y from the input fisheye circular image. The rescaled X and Y are then converted to an address in the input capture

buffer. Data from this address is then output to the encoder for display. The variables Pan, Tilt, and Zoom are the angle of interest commanded by the user input module 290.

Given:

- 5           Pan = Commanded Pan angle in degrees  
          Tilt = Commanded Tilt angle in degrees  
          Zoom = Requested Zoom scale (ie: 1,2,3,etc....)  
          Wx = Output Window pixel x coordinate (-180 to +180 )  
          Wy = Output Window pixel y coordinate ( +120 down to -120 )  
10          IRx = Image buffer fisheye image X axis radius in pixels  
          IRy = Image buffer fisheye image Y axis radius in pixels  
          ICx = Image buffer fisheye image center X coordinate in pixels, as  
              conceptually shown in Figure 4  
          ICy = Image buffer fisheye image center Y coordinate in pixels, as  
15          conceptually shown in Figure 4.

- [0057]     The vertical and horizontal components of the virtual view window (center at origin 0,0) are converted from pixel units to unit values from 0 to less than one by dividing pixel number by the respective image radius. One is the maximum length from the center of the virtual output  
20          image, to the edge of the image. By scaling to the image radius much of the 3-dimensional transformation math can be simplified.

**Equation 11**      $W_x = WindowX / IR_x$

**Equation 12**      $W_y = WindowY / IR_y$

- [0058]     To scale the Wx and Wy components by Zoom, the following  
25          equations are used.

**Equation 13**      $W_x = W_x \times 1/Zoom$

**Equation 14**      $W_y = W_y \times 1/Zoom$

[0059] The desired radius squared is calculated from the vertical and horizontal components using the Pythagorean theorem.

**Equation 15**  $Wr^2 = Wx^2 + Wy^2$

5 [0060] For each pixel, to convert the two dimensional radius squared and scaled components to a three dimensional value with a unit radius of 1, the following equations are used.

Given:

**Equation 16**  $Ix^2 + Iy^2 + Iz^2 = 1$

Then:

10 **Equation 17**  $Iz = \sqrt{1 - Wr^2}$

[0061] Assuming the acquired image center is perpendicular to the Z axis, a rotation of the desired window to a desired spot on the image can be performed using the Pan and Tilt values to simulate a movable dome.

15 [0062] To rotate around the X axis for tilt, the following equations are used.

**Equation 18**  $Iz' = (Iz \times \cos(Tilt)) - (Wy \times \sin(Tilt))$

**Equation 19**  $Wy' = (Wy \times \cos(Tilt)) + (Iz \times \sin(Tilt))$

[0063] To rotate around the Z axis for pan, the following equations are used.

20 **Equation 20**  $Wx' = (Wx \times \cos(Pan)) - (Wy' \times \sin(Pan))$

**Equation 21**  $Wy' = (Wx \times \sin(Pan)) + (Wy' \times \cos(Pan))$

[0064] To convert unit values to Image Buffer pixel units, the following equations are used.

**Equation 22**  $Ix = Wx' \times IRx$

25 **Equation 23**  $Iy = Wy' \times IRy$

[0065] To align the origin relative to upper left of input image buffer, the following equations are used.

**Equation 24**  $I_x = I_x + IC_x$

**Equation 25**  $I_y = I_y + IC_y$

[0066] The image buffer 220 may be addressed as a 1024 byte by 1024 byte array. The memory location, in the image buffer 220, for the pixel data is calculated using the

**Equation 26**  $PixelData = (I_y \times 1024) + I_x$

[0067] PixelData may then be sent to the encoder 235 for output to the display window.

[0068] Advantageously, the panoramic views 110 and 160 provided by the invention provide context for the virtual views 130 and 140. A user never loses sight as to what is happening in the monitored area outside a virtual view, even as the user zooms in on a specific event or object. The user does not have to keep changing video sources to view areas outside the virtual views 130 and 140. Concomitantly, the reference windows 120 and 150 provide a visual indication of the portion of a panoramic view represented by a virtual view. Thus a user can readily determine where in a monitored area they are looking. In an embodiment that includes an additional dome camera system, the ability to switch to the additional camera to produce a high resolution, high zoom factor, virtual view gives the user the ability to resolve small details of a scene.

[0069] Referring now to **Figure 5**, a high level flow chart for a process of creating a video user interface in accordance with an exemplary implementation of the invention is shown. The steps may be implemented using a system with a programmed processor as described above. **Figure 5** may equally represent a high level block diagram of an exemplary system constructed in accordance with the invention, implementing the steps thereof.

[0070] In step 510, wide-angle data (e.g., fisheye lens image data) is buffered. The data may be supplied from a video camera system with a decoder or from a source of pre-recorded data.

5 [0071] Next, in step 520, pan, tilt and zoom angles are determined to specify regions of interest in the panoramic views for display in virtual views. Pan, tilt and zoom angles may be supplied by a programmed algorithm, a table or a user input module. Those skilled in the art will appreciate that values other than pan, tilt and zoom (e.g., Cartesian coordinate values) may be specified to define areas of interest in the panoramic views.

10 [0072] Next, in step 530, reference data for reference windows are determined. The reference data preferably corresponds to rectangular windows overlaying the virtual view regions of interest within the panoramic views. Preferably, the rectangular windows move according to a programmed algorithm, a table or user commands from a user input module. As the rectangular windows move, the virtual views correspondingly change.

15 [0073] Next, in step 540, buffered wide-angle data are transformed using a panoramic transformation algorithm for panoramic view data, and using a virtual view transformation algorithm for virtual view data.

20 [0074] Next, in step 550, the transformed data and reference data are encoded for output. The transformed data defines any other virtual views and the panoramic views. The reference data defines the reference windows. The encoded output may be sent to a display monitor and/or to a recorder.

25 [0075] Those skilled in the art will appreciate that the order in which some of the steps may vary from the order shown in the flowchart without departing from the scope of the invention. For example, reference data



may be determined before, after or during transformation of the buffered data. Additionally, a pre-programmed pan, tilt and zoom may be determined before wide-angle data is initially buffered.

5 [0076] Referring now to **Figure 6**, a high level flow chart for another process of creating video user interface in accordance with an exemplary implementation of the invention is shown. As with the process illustrated by the flowchart of Figure 5, the steps may be implemented using a system with a programmed processor as described above. **Figure 6** may equally represent a high level block diagram of an exemplary system constructed  
10 in accordance with the invention, implementing the steps thereof.

[0077] In step **610**, wide-angle data (e.g., fisheye lens image data) is buffered. The data may be supplied from a video camera system with a decoder or from a source of pre-recorded data.

15 [0078] Next, in step **620**, pan, tilt and zoom angles are determined to specify regions of interest in the panoramic views for display in virtual views. Pan, tilt and zoom angles may be supplied by a programmed algorithm, a table or a user input module. Those skilled in the art will appreciate that values other than pan, tilt and zoom (e.g., Cartesian coordinate values) may be specified to define areas of interest in the  
20 panoramic views.

[0079] Next, in step **630**, reference data for reference windows are determined. The reference data preferably corresponds to rectangular windows overlaying the virtual view regions of interest within the panoramic views. Preferably, the rectangular windows move according to  
25 a programmed algorithm, a table or user commands from a user input module. As the rectangular windows move, the virtual views correspondingly change.

[0080] Next, in step 640, position data is communicated to a dome camera system. The communicated position data will cause the dome camera to aim at and zoom to the region of the monitored area corresponding to the determined pan, tilt and zoom values.

5 [0081] In step 650, dome camera data is buffered. The buffered data may be encoded, without transformation, for output to a display monitor and/or to a recorder as a virtual view.

[0082] Next, in step 660, buffered wide-angle data are transformed using a panoramic transformation algorithm for panoramic view data, and using  
10 a virtual view transformation algorithm for virtual view data.

[0083] Next, in step 670, buffered dome camera data are obtained for encoding. The dome camera data may be obtained from a dome camera buffer by a processor and communicated with transformed wide-angle data for encoding.

15 [0084] Next, in step 680, the transformed data, dome camera data and reference data are encoded for output. The dome camera data defines one virtual view. The transformed data defines any other virtual views and the panoramic views. The reference data defines the reference windows. The encoded output may be sent to a display monitor and/or to a recorder.

20 [0085] Those skilled in the art will appreciate that the order in which some of the steps are performed may not be important. For example, the reference data may be determined before, after or during transformation of the buffered data. Additionally, by way of example, a pre-programmed pan, tilt and zoom may be determined before the wide-angle data is  
25 initially buffered. As another example, dome position data may be sent before reference data is determined. As yet another example, dome data may be buffered after transformation of wide-angle data. Also, dome camera data may be obtained by the processor before completion of

transformation. Such variations in order are intended to come within the scope of the invention.

[0086] Although the foregoing description is directed to preferred embodiments of the invention, it is noted that other variations and  
5 modifications will be apparent to those skilled in the art, and may be made without departing from the spirit or scope of the invention.